

## AMENDMENTS TO THE CLAIMS

1. (Currently Amended) A method comprising:  
electing a first server as active manager server, wherein the first server resides in  
a chassis;  
determining automatically, by receiving an indication, if the first server has failed  
or has been overloaded, wherein the indication is generated based on  
health matrices and performance matrices; ~~and~~  
electing a second server automatically as the active manager server to replace the  
first server as the active manager server in response to the indication  
received, wherein the second server resides in the chassis; and  
redirecting requests for the first server to the second server.
2. (Original) The method of claim 1, wherein the election is performed based on a  
predetermined criteria, wherein the predetermined criteria comprises electing a  
server with the lowest IP address as the active manager server.
3. (Previously Presented) The method of claim 1, further comprising:  
extracting the health metrics and performance metrics, wherein the health metrics  
and performance metrics are dynamic;  
replicating the health metrics and performance metrics, wherein the replicating  
the health metrics and performance metrics is performed periodically; and  
dynamically updating a database populated with the health metrics and  
performance metrics.
4. (Original) The method of claim 3, wherein the health metrics are server-based.
5. (Original) The method of claim 3, wherein the health metrics comprise tracking  
power levels and temperature levels based on predetermined thresholds.

6. (Original) The method of claim 3, wherein the performance metrics comprise operating system-based metrics, kernel-based metrics, and server-based metrics.
7. (Previously Presented) The method of claim 3, wherein the performance metrics comprise tracking CPU utilization and memory utilization based on predetermined thresholds.
8. (Original) The method of claim 3, further comprises an alert mechanism to alert whenever the health metrics or the performance metrics violate the predetermined thresholds.
9. (Original) The method of claim 3, further comprising replicating identification information, wherein the identification information is static.
10. (Currently Amended) A high-availability management system comprising:
  - a chassis comprising a plurality of slots;
  - a plurality of server modules coupled ~~with~~ to the plurality of slots, wherein a first server module of the plurality of server modules is elected an active manager server;
  - an indication to automatically determine if the first server module has failed or has been overloaded, wherein the indication is generated based on health matrices and performance matrices; ~~and~~
  - a second server module to automatically replace the first server module as the active manager server in response to the indication received; and
  - a redirection process to redirect requests for the first server module to the second server module.
11. (Original) The high-availability management system of claim 10, further comprising a database coupled to the chassis for storing information regarding chassis identification, slot identification, and server module type.

12. (Original) The high-availability management system of claim 10, wherein the first server module of the plurality of server modules is elected the active manager server based on a predetermined criteria.
13. (Original) The high-availability management system of claim 10, wherein a second server module of the plurality of server modules is elected the active manager server, based on the predetermined criteria, to replace the first server module as the active manager server when the first server module is to be replaced.
14. (Currently Amended) The high-availability management system of claim 10, wherein the election of the first server module as the active manager server is performed by middleware, wherein the middleware ~~is~~ comprises a software.
15. (Currently Amended) The high-availability management system of claim 13, wherein the election of the second server module as the active manager server is performed by the middleware, ~~wherein the middleware is a software.~~
16. (Original) The high-availability management system of claim 10, wherein the first server module is elected from a group comprising servers, telephone line cards, and power substations.
17. (Currently Amended) A method of uninterrupted management using sticky identification comprising:  
  
assigning a chassis identification to a chassis coupled to a computer, wherein the  
  
chassis comprises a slot;  
  
assigning a slot identification to the slot based on the slot's location in the chassis;

assigning a server module type to the slot based on the chassis identification and  
 the slot identification, wherein the server module type indicates server  
 module characteristics;  
 electing a first server module as active manager server, wherein the first server  
 module resides in the chassis;  
 determining automatically, by receiving an indication, if the first server module  
 has failed or has been overloaded, wherein the indication is generated  
 based on health matrices and performance matrices; ~~and~~  
 electing a second server module automatically as the active manager server to  
 replace the first server module as the active manager server in response to  
 the indication received, wherein the second server module resides in the  
 chassis; and  
redirecting requests for the first server module to the second server module.

18. (Original) The method of uninterrupted management using sticky identification of claim 17, further comprising retaining the server module characteristics corresponding to the server module type.
19. (Original) The method of uninterrupted management using sticky identification of claim 17, further comprising:  
 removing a first server module from the slot;  
 coupling a second server module to the slot; and  
 managing the second server module based on the server module characteristics corresponding to the server module type, wherein the managing the second server module is performed without updating a network management system.

20. (Original) The method of uninterrupted management using sticky identification of claim 17, further comprising:
- assigning a user-defined chassis identification;
  - assigning a user-defined slot identification;
  - assigning a user-defined module identification; and
  - retaining the user-defined chassis identification and the user-defined slot identification and the user-defined module identification.
21. (Currently Amended) A machine-readable medium having stored thereon data representing sets of instructions, the sets of instructions which, when executed by a machine, cause the machine to:
- elect a first server as active manager server, wherein the first server resides in a chassis;
  - determine automatically, by receiving an indication, if the first server failed or is overloaded, wherein the indication is generated based on health matrices and performance matrices; ~~and~~
  - elect a second server automatically as the active manager server to replace the first server as the active manager server in response to the indication received, wherein the second server resides in the chassis; and
  - redirect requests for the first server to the second server.
22. (Original) The machine-readable of claim 21, wherein the election is performed based on a predetermined criteria, wherein the predetermined criteria comprises electing a server with the lowest IP address as the active manager server.

23. (Previously Presented) A machine-readable medium of claim 21, wherein the sets of instructions which, when executed by the machine, further cause the machine to:
- extract the health metrics and performance metrics, wherein the health metrics and performance metrics are dynamic;
- replicate the health metrics and performance metrics, wherein the replicating the health metrics and performance metrics is performed periodically; and
- dynamically update a database populated with the health metrics and performance metrics.
24. (Currently Amended) A machine-readable medium having stored thereon data representing sets of instructions, the sets of instructions which, when executed by a machine, cause the machine to:
- assign a chassis identification to a chassis coupled to a computer, wherein the chassis comprises a slot;
- assign a slot identification to the slot based on the slot's location in the chassis;
- assign a server module type to the slot based on the chassis identification and the slot identification, wherein the server module type indicates server module characteristics;
- elect a first server module as active manager server, wherein the first server module resides in the chassis;
- determine automatically, by receiving an indication, if the first server module has failed or has been overloaded, wherein the indication is generated based on health matrices and performance matrices; and

elect a second server module automatically as the active manager server to replace the first server module as the active manager server in response to the indication received, wherein the second server module resides in the chassis; and  
redirect requests for the first server to the second server.

25. (Previously Presented) The machine-readable medium of claim 24, wherein the sets of instructions which, when executed by the machine, further cause the machine to retain the server module characteristics corresponding to the server module type.
26. (Previously Presented) The machine-readable medium of claim 24, wherein the sets of instructions which, when executed by the machine, further cause the machine to:
- remove a first server module from the slot;
  - couple a second server module to the slot; and
  - manage the second server module based on the server module characteristics corresponding to the server module type, wherein the managing the second server module is performed without updating a network management system.